AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior version, and listings, of claims in the application:

Listing of Claims:

1-41. (Cancelled)

42. (Previously Presented) A graphics system comprising a two-dimensional graphics imaging pipeline constructed and arranged to manipulate two-dimensional (2D) images represented by pixel data comprising color and X,Y coordinate data, and excluding Z coordinate data, and to composite separately generated three-dimensional (3D) images represented by pixel data comprising X,Y,Z coordinate and color data,

wherein the X,Y coordinate data define horizontal and vertical dimensions of a pixel's display screen location, and wherein the Z coordinate defines an orthogonal distance from viewpoint to the image rendered at a pixel.

43. (Previously Presented) The graphics system of claim 42, wherein the graphics system comprises a rendering pipeline comprising:

a geometric pipeline constructed and arranged to generate a two-dimensional image from one or more model views represented by primitive data; and

said imaging pipeline.

44. (Previously Presented) The graphics system of claim 43, wherein operational components of said geometric pipeline are utilized by said imaging pipeline to composite said separately generated 3D images.

45. (Previously Presented) The graphics system of claim 42,

wherein the graphics system further comprises a frame buffer for storing pixel data to be displayed on a display device, and

wherein said 3D images comprise:

a stored image stored in the frame buffer; and

a next image to be composited with the stored image.

46. (Previously Presented) The graphics system of claim 45, wherein said next image is stored in a system memory.

47. (Previously Presented) The graphics system of claim 45, wherein said next image is stored in said frame buffer.

48. (Previously Presented) The graphics system of claim 42, wherein said imaging pipeline comprises:

a depth buffer configured to store Z coordinate data for each pixel in a display scene; and

a depth test module constructed and arranged to compare Z coordinate data of said 3D images, and to store in said depth buffer Z coordinate data of each pixel of the 3D image that is closest to viewpoint.

49. (Previously Presented) The graphics system of claim 48,

wherein for each pixel, said imaging pipeline writes to a frame buffer of the graphics system the color data of the 3D image that is closest to the viewpoint.

50. (Previously Presented) The graphics system of claim 42, wherein said imaging pipeline receives said Z coordinate data over a data channel of the imaging pipeline configured to transfer data other than Z coordinate data, and receives said X,Y coordinate data over an address data channel.

51. (Previously Presented) The graphics system of claim 50, wherein the data other than Z coordinate data is color data, and the channel provided to transfer data other than Z coordinate data is a color data channel.

52. (Previously Presented) The graphics system of claim 42, wherein the graphics system further comprises a frame buffer for storing pixel data, and wherein the 3D images comprise a stored 3D image stored in the frame buffer and a next 3D image to be composited with the stored 3D image, wherein the next 3D image is passed through the imaging pipeline twice to composite it with the stored 3D image.

53. (Previously Presented) The graphics system of claim 48

wherein said depth test module receives Z coordinate data of a next 3D image to be compared with a 3D image stored in a frame buffer of the graphics system over a color data channel of the imaging pipeline.

54. (Previously Presented) The graphics system of claim 48, wherein said indication of which 3D image is closest to the viewpoint at each pixel is provided through the setting of a corresponding bit in a stencil buffer of the imaging pipeline.

55. (Previously Presented) The graphics system of claim 48, wherein said imaging pipeline further comprises:

a stencil buffer containing stencil bits for each pixel in the display scene; and

a stencil test module constructed and arranged to set said stencil bits to indicate which 3D image is closest to the viewpoint based on the results of the depth test.

56. (Previously Presented) The graphics system of claim 52, wherein in an initial pass through the imaging pipeline, a depth test is performed to determine which 3D image is to be rendered at each pixel, with an indication of that 3D image and its Z coordinate data being stored in a memory location associated with each pixel, and in a subsequent pass through the imaging pipeline, storing in the frame buffer color data of the 3D image which is to be rendered at each pixel based on the indication stored during the initial pass through the imaging pipeline.

57. (Previously Presented) A method for compositing three-dimensional (3D) images in a two-dimensional (2D) imaging pipeline configured to manipulate 2D images represented by pixel data comprising color and X,Y coordinate data, and excluding Z coordinate data, the method comprising:

storing in a frame buffer a stored 3D image including color data and X,Y,Z coordinate data;

processing in the 2D imaging pipeline Z coordinate data of a next 3D image to determine whether the stored or next 3D image is to be rendered at each pixel in a resulting composited image; and

replacing said stored color data with color data of said next 3D image for each pixel at which the next 3D image is to be rendered in the composited image,

wherein the X,Y coordinate data define horizontal and vertical dimensions of a pixel's display screen location, and wherein the Z coordinate defines an orthogonal distance from viewpoint to the image rendered at a pixel.

58. (Previously Presented) The method of claim 578, wherein said processing Z coordinate data comprises:

transferring Z coordinate data of the next image through an available data channel of imaging pipeline;

depth testing the stored and next images;

updating a depth buffer as necessary to store Z coordinate data of an image that is closest to a current viewpoint; and

recording an indication of which 3D image is the closest image.

59. (Previously Presented) The method of claim 58, wherein said available data channel is a color data channel.

60. (Previously Presented) The method of claim 58, wherein the imaging pipeline consists of a color data channel and an address data channel, and wherein said transferring Z coordinate data comprises:

selecting a color data channel of the imaging pipeline to provide the next image Z coordinate data for the depth testing; and

transferring the next image Z coordinate data over the color data channel.

61. (Previously Presented) The method of claim 60, wherein said transferring Z coordinate data further comprises:

masking data writes to the color buffer to prevent the next image Z coordinate data from being written to the color buffer.

62. (Previously Presented) The method of claim 58, wherein said recording comprises:

performing a stencil test such that a stencil buffer is modified to reflect whether the stored or next image is closest to the viewpoint.

63. (Previously Presented) The method of claim 57, wherein the imaging pipeline consists of a color data channel and an address data channel, and wherein replacing the stored color data comprises:

receiving over the color data channel the color data of the next 3D image, and

storing the color data of the next 3D image when the recording step records an indication that the next 3D image is the closest 3D image.

64. (Previously Presented) The method of claim 57, wherein said storing comprises:

generating the stored image;

storing the stored image in system memory; and

transferring the stored image from the system memory to the frame buffer.

65. (Previously Presented) The method of claim 57, wherein replacing the stored color data comprises:

transferring the image color data over color data channel;

performing stencil test such that the stencil test passes for each pixel of the next image that is closer to the viewpoint that the stored image; and

updating the color buffer in accordance with the stencil in stencil buffer thereby resulting in values in the color buffer that correspond to pixels that passed the depth testing.

66. (Previously Presented) A method for compositing a stored and a next three-dimensional image in an imaging/two-dimensional graphics pipeline configured to manipulate two-dimensional images represented by pixel data comprising X,Y coordinate data defining horizontal and vertical dimensions of a pixel's display screen location and color data, and excluding Z coordinate data defining an orthogonal distance from the viewpoint to the image rendered at a pixel, the method comprising the steps of:

storing the stored 3D image in a frame buffer of the imaging pipeline, wherein said stored 3D image includes color data and X,Y,Z coordinate data;

processing successively potions of the next 3D image through the imaging pipeline to select which of the next or stored 3D image is closest to a viewpoint; and

saving to the frame buffer color data of the selected 3D image.

67. (Previously Presented) A graphics system comprising a two-dimensional imaging pipeline configured to manipulate two-dimensional (2D) images and to composite a separately-generated three-dimensional (3D) image stored in a frame buffer, and a next 3D image, comprising,

a color data channel adapted to receive Z coordinate data and color data of a next 3D image;

an image compositing module configured to perform a depth test to determine which 3D image is to be rendered at each pixel based on Z coordinate data of the next image received over the color data channel, and Z coordinate data of the stored 3D image, and to store the Z coordinate data of the 3D image to be rendered at that pixel in a depth buffer, and a stencil test to form a stencil mask identifying which 3D image is the image that is to be rendered at each pixel,

wherein the imaging pipeline, in response to receipt of color data over the color data channel, updates a color buffer to have stored therein color data of the 3D image to be rendered at each pixel of the composite image.

68. (Previously Presented) The graphics system of claim 67, wherein said two-dimensional images and said three-dimensional images are represented by pixel data.

69. (Previously Presented) The graphics system of claim 67, wherein the graphics system comprises a rendering pipeline comprising:

a geometric pipeline constructed and arranged to create a two-dimensional image from primitive data; and

said imaging pipeline.